# Programación Python para BigData

## Lección 1: Introducción



eip ESCUELA INTERNACIONAL POSGRADOS

# Indice

# Introducción

En la presente lección se han presentado una serie de herramientas que se emplearán mas adelante a medida que se desarrolle la asignatura.

Por el momento se pide instalar Docker y Docker Compose adjuntando las debida capturas de pantalla así como dar respuesta a una serie de preguntas.

# Primera parte

## Instalación de los programas base

### Docker

En el presente apartado se procederá a la instalación de Docker siguiendo las instrucciones del manual:

1.- Instalamos la librerias ecesarias:

```
elidas@FireBall:~$ sudo apt-get install apt-transport-https ca-certificates curl
 gnupg lsb-release
[sudo] contraseña para elidas:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
lsb-release ya está en su versión más reciente (9.20170808ubuntu1).
ca-certificates ya está en su versión más reciente (20210119~18.04.1).
curl ya está en su versión más reciente (7.58.0-2ubuntu3.14).
gnupg ya está en su versión más reciente (2.2.4-1ubuntu1.4).
apt-transport-https ya está en su versión más reciente (1.6.14).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
```

```
elidas@FireBall:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sud
o gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
gpg: AVISO: propiedad insegura del directorio personal '/home/elidas/.gnupg'
```

2.- Actualizamos los repositorios

```
elidas@FireBall:~$ echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-ar
chive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) s
table" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
elidas@FireBall:~$ sudo apt update
Des:1 https://download.docker.com/linux/ubuntu bionic InRelease [64,4 kB]
Des:2 http://security.ubuntu.com/ubuntu bionic-security InRelease [88,7 kB]
Obj:3 http://es.archive.ubuntu.com/ubuntu bionic InRelease
Des:4 http://es.archive.ubuntu.com/ubuntu bionic-updates InRelease [88,7 kB]
Des:5 https://download.docker.com/linux/ubuntu bionic/stable amd64 Packages [19,
8 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu bionic-backports InRelease [74,6 kB]
Descargados 336 kB en 2s (190 kB/s)
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Todos los paquetes están actualizados.
```

3.- Instalamos Docker

```
elidas@FireBall:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  docker-ce-rootless-extras docker-scan-plugin pigz
Paquetes sugeridos:
  aufs-tools cgroupfs-mount | cgroup-lite
Paquetes recomendados:
  slirp4netns
Se instalarán los siguientes paquetes NUEVOS:
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras
  docker-scan-plugin pigz
0 actualizados, 6 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 96,6 MB de archivos.
Se utilizarán 406 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

```
Des:1 http://es.archive.ubuntu.com/ubuntu bionic/universe amd64 pigz amd64 2.4-1
 [57,4 kB]
Des:2 https://download.docker.com/linux/ubuntu bionic/stable amd64 containerd.io
 amd64 1.4.9-1 [24,7 MB]
Des:3 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-ce-cli
 amd64 5:20.10.8~3-0~ubuntu-bionic [38,8 MB]
Des:4 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-ce amd
64 5:20.10.8~3-0~ubuntu-bionic [21,2 MB]
Des:5 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-ce-roo
tless-extras amd64 5:20.10.8~3-0~ubuntu-bionic [7.911 kB]
Des:6 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-scan-p
lugin amd64 0.8.0~ubuntu-bionic [3.888 kB]
Descargados 96,6 MB en 5s (17,7 MB/s)
Seleccionando el paquete pigz previamente no seleccionado.
(Leyendo la base de datos ... 182901 ficheros o directorios instalados actualmen
te.)
Preparando para desempaquetar .../0-pigz_2.4-1_amd64.deb ...
Desempaquetando pigz (2.4-1) ...
Seleccionando el paquete containerd.io previamente no seleccionado.
Preparando para desempaquetar .../1-containerd.io_1.4.9-1_amd64.deb ...
Desempaquetando containerd.io (1.4.9-1) ...
Seleccionando el paquete docker-ce-cli previamente no seleccionado.
Preparando para desempaquetar .../2-docker-ce-cli_5%3a20.10.8~3-0~ubuntu-bionic_
amd64.deb ...
Desempaquetando docker-ce-cli (5:20.10.8~3-0~ubuntu-bionic) ...
Seleccionando el paquete docker-ce previamente no seleccionado.
Preparando para desempaquetar .../3-docker-ce_5%3a20.10.8~3-0~ubuntu-bionic_amd6
4.deb ...
Desempaquetando docker-ce (5:20.10.8~3-0~ubuntu-bionic) ...
Seleccionando el paquete docker-ce-rootless-extras previamente no seleccionado.
Preparando para desempaquetar .../4-docker-ce-rootless-extras_5%3a20.10.8~3-0~ub
untu-bionic_amd64.deb ...
Desempaquetando docker-ce-rootless-extras (5:20.10.8~3-0~ubuntu-bionic) ...
Seleccionando el paquete docker-scan-plugin previamente no seleccionado.
Preparando para desempaquetar .../5-docker-scan-plugin_0.8.0~ubuntu-bionic_amd64
.deb ...
Desempaquetando docker-scan-plugin (0.8.0~ubuntu-bionic) ...
Configurando containerd.io (1.4.9-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service →
 /lib/systemd/system/containerd.service.
Configurando docker-ce-rootless-extras (5:20.10.8~3-0~ubuntu-bionic) ...
Configurando docker-scan-plugin (0.8.0~ubuntu-bionic) ...
Configurando docker-ce-cli (5:20.10.8~3-0~ubuntu-bionic) ...
Configurando pigz (2.4-1) ...
Configurando docker-ce (5:20.10.8~3-0~ubuntu-bionic) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /li
b/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/sy
stemd/system/docker.socket.
Procesando disparadores para systemd (237-3ubuntu10.51) ...
Procesando disparadores para man-db (2.8.3-2ubuntu0.1) ...
Procesando disparadores para ureadahead (0.100.0-21) ...
```

4.- Verificamos la instalación:

```
elidas@FireBall:~$ sudo docker --version
Docker version 20.10.8, build 3967b7d
```

5.- Comprobamos que se ejecuta bien una imagen:

```
elidas@FireBall:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
b8dfde127a29: Pull complete
Digest: sha256:7d91b69e04a9029b99f3585aaaccae2baa80bcf318f4a5d2165a9898cd2dc0a1
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```

6.- Verificamos que el servicio activo de docker y de containerd funciona

```
elidas@FireBall:~$ sudo systemctl status containerd.service
● containerd.service - containerd container runtime
   Loaded: loaded (/lib/systemd/system/containerd.service; enabled; vendor prese
   Active: active (running) since Thu 2021-09-02 20:00:57 CEST; 5min ago
     Docs: https://containerd.io
 Main PID: 7421 (containerd)
    Tasks: 10
   CGroup: /system.slice/containerd.service
           └─7421 /usr/bin/containerd

sep 02 20:00:57 FireBall containerd[7421]: time="2021-09-02T20:00:57.900490783+0
sep 02 20:00:57 FireBall containerd[7421]: time="2021-09-02T20:00:57.900505049+0
sep 02 20:00:57 FireBall containerd[7421]: time="2021-09-02T20:00:57.900513040+0
sep 02 20:00:57 FireBall containerd[7421]: time="2021-09-02T20:00:57.901086126+0
sep 02 20:00:57 FireBall containerd[7421]: time="2021-09-02T20:00:57.901208175+0
sep 02 20:00:57 FireBall containerd[7421]: time="2021-09-02T20:00:57.901319420+0
sep 02 20:00:57 FireBall systemd[1]: Started containerd container runtime.
sep 02 20:02:55 FireBall containerd[7421]: time="2021-09-02T20:02:55.008149847+0
sep 02 20:02:55 FireBall containerd[7421]: time="2021-09-02T20:02:55.882661624+0
sep 02 20:02:55 FireBall containerd[7421]: time="2021-09-02T20:02:55.882816624+0
lines 1-19/19 (END)
```

```
elidas@FireBall:~$ sudo systemctl status docker.service
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: e
   Active: active (running) since Thu 2021-09-02 20:01:05 CEST; 2min 45s ago
     Docs: https://docs.docker.com
 Main PID: 7576 (dockerd)
    Tasks: 10
   CGroup: /system.slice/docker.service
           └─7576 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/contain

sep 02 20:01:03 FireBall dockerd[7576]: time="2021-09-02T20:01:03.655756434+02:0
sep 02 20:01:03 FireBall dockerd[7576]: time="2021-09-02T20:01:03.655777704+02:0
sep 02 20:01:03 FireBall dockerd[7576]: time="2021-09-02T20:01:03.657349587+02:0
sep 02 20:01:04 FireBall dockerd[7576]: time="2021-09-02T20:01:04.485059785+02:0
sep 02 20:01:05 FireBall dockerd[7576]: time="2021-09-02T20:01:05.146242099+02:0
sep 02 20:01:05 FireBall dockerd[7576]: time="2021-09-02T20:01:05.701230394+02:0
sep 02 20:01:05 FireBall dockerd[7576]: time="2021-09-02T20:01:05.703062819+02:0
sep 02 20:01:05 FireBall systemd[1]: Started Docker Application Container Engine
sep 02 20:01:05 FireBall dockerd[7576]: time="2021-09-02T20:01:05.987607044+02:0
sep 02 20:02:55 FireBall dockerd[7576]: time="2021-09-02T20:02:55.880832776+02:0
lines 1-19/19 (END)
```

7.- Habilitamos la posibilidad de que docker siempre este activo, aunque se reinicie el ordenador

```
elidas@FireBall:~$ sudo systemctl enable docker.service
Synchronizing state of docker.service with SysV service script with /lib/systemd
/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable docker
```

Falta la habilitación de containerd.service porque corte de mas la imagen.

## Docker-compose

De entrada se intentó instalar docker compose siguiendo los pasos indicados en la leccion:

1.- Obtención e instalación del paquete:

```
elidas@FireBall:~$ sudo apt-get install docker-compose
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  golang-docker-credential-helpers libpython-stdlib python python-asn1crypto
  python-backports.ssl-match-hostname python-cached-property python-certifi
  python-cffi-backend python-chardet python-cryptography python-docker
  python-dockerpty python-dockerpycreds python-docopt python-enum34
  python-funcsigs python-functools32 python-idna python-ipaddress
  python-jsonschema python-minimal python-mock python-openssl python-pbr
  python-pkg-resources python-requests python-six python-texttable
  python-urllib3 python-websocket python-yaml python2.7 python2.7-minimal
Paquetes sugeridos:
  python-doc python-tk python-cryptography-doc python-cryptography-vectors
  python-enum34-doc python-funcsigs-doc python-mock-doc python-openssl-doc
  python-openssl-dbg python-setuptools python-socks python-ntlm python2.7-doc
  binfmt-support
Paquetes recomendados:
  docker.io
Se instalarán los siguientes paquetes NUEVOS:
  docker-compose golang-docker-credential-helpers libpython-stdlib python
  python-asn1crypto python-backports.ssl-match-hostname python-cached-property
  python-certifi python-cffi-backend python-chardet python-cryptography
  python-docker python-dockerpty python-dockerpycreds python-docopt
  python-enum34 python-funcsigs python-functools32 python-idna
  python-ipaddress python-jsonschema python-minimal python-mock python-openssl
```

```
Configurando python-six (1.11.0-2) ...
Configurando python-dockerpty (0.4.1-1) ...
Configurando python-pbr (3.1.1-3ubuntu3) ...
update-alternatives: utilizando /usr/bin/python2-pbr para proveer /usr/bin/pbr (
pbr) en modo automático
Configurando python-enum34 (1.1.6-2) ...
Configurando python-funcsigs (1.0.2-4) ...
Configurando python-docopt (0.6.2-1build1) ...
Configurando python-ipaddress (1.0.17-1) ...
Configurando python-cached-property (1.3.1-1) ...
Configurando python-urllib3 (1.22-1ubuntu0.18.04.2) ...
Configurando python-chardet (3.0.4-1) ...
Configurando python-dockerpycreds (0.2.1-1) ...
Configurando python-mock (2.0.0-3) ...
Configurando python-websocket (0.44.0-0ubuntu2) ...
update-alternatives: utilizando /usr/bin/python2-wsdump para proveer /usr/bin/ws
dump (wsdump) en modo automático
Configurando python-cryptography (2.1.4-1ubuntu1.4) ...
Configurando python-requests (2.18.4-2ubuntu0.1) ...
Configurando python-jsonschema (2.6.0-2) ...
update-alternatives: utilizando /usr/bin/python2-jsonschema para proveer /usr/bi
n/jsonschema (jsonschema) en modo automático
Configurando python-openssl (17.5.0-1ubuntu1) ...
Configurando python-docker (2.5.1-1) ...
Configurando docker-compose (1.17.1-2) ...
Procesando disparadores para gnome-menus (3.13.3-11ubuntu1.1) ...
Procesando disparadores para mime-support (3.60ubuntu1) ...
Procesando disparadores para desktop-file-utils (0.23-1ubuntu3.18.04.2) ...
Procesando disparadores para man-db (2.8.3-2ubuntu0.1) ...
```

2.- Verificación de la versión:

```
elidas@FireBall:~$ sudo docker-compose --version
Traceback (most recent call last):
  File "/usr/bin/docker-compose", line 6, in <module>
    from pkg_resources import load_entry_point
  File "/usr/lib/python3/dist-packages/pkg_resources/__init__.py", line 3088, in
 <module>
    @_call_aside
  File "/usr/lib/python3/dist-packages/pkg_resources/__init__.py", line 3072, in
 _call_aside
    f(*args, **kwargs)
  File "/usr/lib/python3/dist-packages/pkg_resources/__init__.py", line 3101, in
 _initialize_master_working_set
    working_set = WorkingSet._build_master()
  File "/usr/lib/python3/dist-packages/pkg_resources/__init__.py", line 574, in
_build_master
    ws.require(__requires__)
  File "/usr/lib/python3/dist-packages/pkg_resources/__init__.py", line 892, in
require
    needed = self.resolve(parse_requirements(requirements))
  File "/usr/lib/python3/dist-packages/pkg_resources/__init__.py", line 778, in
resolve
    raise DistributionNotFound(req, requirers)
pkg_resources.DistributionNotFound: The 'docker-compose==1.17.1' distribution wa
s not found and is required by the application
```

Al realizar este paso se detectó un problema, el paquete no se habia instalado bien, para solucionar dicho problema se recurre a la pagina oficial de Docker donde se especifican los siguientes pasos:

1.- Desinstalar lo instalado

```
elidas@FireBall:~$ sudo apt remove docker-compose
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no
son necesarios.
  golang-docker-credential-helpers libpython-stdlib python python-asn1crypto
  python-backports.ssl-match-hostname python-cached-property python-certifi
  python-cffi-backend python-chardet python-cryptography python-docker
  python-dockerpty python-dockerpycreds python-docopt python-enum34
  python-funcsigs python-functools32 python-idna python-ipaddress
  python-jsonschema python-minimal python-mock python-openssl python-pbr
  python-pkg-resources python-requests python-six python-texttable
  python-urllib3 python-websocket python-yaml python2.7 python2.7-minimal
Utilice «sudo apt autoremove» para eliminarlos.
Los siguientes paquetes se ELIMINARÁN:
  docker-compose
0 actualizados, 0 nuevos se instalarán, 1 para eliminar y 0 no actualizados.
Se liberarán 517 kB después de esta operación.
¿Desea continuar? [S/n] s
(Leyendo la base de datos ... 184034 ficheros o directorios instalados actualmen
te.)
Desinstalando docker-compose (1.17.1-2) ...
Procesando disparadores para man-db (2.8.3-2ubuntu0.1) ...
```

2.- Instalar ciertas dependencias

```
elidas@FireBall:~$  sudo curl -L "https://github.com/docker/compose/releases/dow
nload/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-co
mpose
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   633  100   633    0     0   3767      0 --:--:-- --:--:-- --:--:--  3745
100 12.1M  100 12.1M    0     0  14.3M      0 --:--:-- --:--:-- --:--:-- 14.3M
```

3.-Modificar el archivo binario y verificar la instalación.

```
elidas@FireBall:~$  sudo chmod +x /usr/local/bin/docker-compose
elidas@FireBall:~$ sudo docker-compose --version
docker-compose version 1.29.2, build 5becea4c
```

# Segunda Parte

Tras realizar la búsqueda requerida en el ejercicio, he hallado tres paginas web que concuerdan en la definición de BigData:

- https://www.powerdata.es/big-data
- https://www.oracle.com/es/big-data/what-is-big-data/
- https://aws.amazon.com/es/big-data/what-is-big-data/

Las tres webs concuerdan en definir el BigData como aquellos datos o conjunto de los mismos cuya variedad y volumen evolucionan a tal velocidad que hace imposible su manejo por medio de métodos ordinarios por el contrario, es necesario tratarlos con métodos que permitan simplificar y reducir la información.

Teniendo en cuenta esta definición y habiendo leído algún otro artículo a fin de hallar mas información, se puede concluir que realmente no tiene sentido hablar de tamaños para clasificar conjuntos de datos como BigData, ya que por grande que sea el conjunto, si este no evoluciona, tarde o temprano, podremos analizar toda la información de este en cambio, si el conjunto evoluciona, y aumenta de tamaño, resulta impensable emplear métodos comunes para su análisis y podemos clasificar al mismo como BigData.

# Tercera parte

*Spotify escucha tus datos*

*https://www.merkleinc.com/es/es/blog/spotify-escucha-datos*

La Web Merkle en su artículo 'Spotify escucha tus datos' relata entre otras cosas, que datos almacena Spotify sobre sus canciones y que muestra a cada usuario o tipo de usuario además de, como emplea dichos datos para ofrecer una mejor experiencia a sus usuarios. Si bien es cierto que no menciona la cantidad de información que almacena, leyendo el artículo se sobreentiende que es mucha y muy variada.

*Big Data: ¿Cuánta información maneja Facebook cada día?*

*https://www.muycomputer.com/2012/08/23/big-data-cuanta-informacion-maneja-facebook-cada-dia/*

En el presente articulo de la pagiuna Muy Computer se habla de la cantida de información que los servidores de facebook manejan cada dia, si bien el articulo es del 2012 nos da la posibilidad de hacernoas una ligera idea de la información que se puede llegar a manejar hoy en dia ya que en este año se habla de 105 Tbytes de información ala hora.

*¿Qué tanto sabe Facebook sobre ti? y cómo puedes verlo y editarlo*

*https://www.mercatitlan.com/blog/que-tanto-sabe-facebook-sobre-ti-y-como-puedes-verlo-y-editarlo*

En este caso, Mercatitlán habla de la información que a fecebook le interesa sobre nosotros, la cual divide en tres categorías principales: intereses, interacciones y datos. Comenta de forma breve cada una de estas categorias dando a ntender que facebook es capaz de conocer casi cada aspecto de nuestro dia a dia, ademas de esllo muestra un ejemplo en el que facebook ha conseguido clasificar los gustos e intereses del escritor casi a la perfeccion.

Teniendo en cuenta los dos ultimos articulos podemos entrender que la cantidad de informacion que generamos en el dia a dia es inmensa y que empresas como facebok han podido almacenarla, cribarla y analizarla gracias a los metodos de analisis del BigData.

## Extra

Por ultimo, durante la búsqueda de la información requerida en los ejercicios anteriores, he encontrado un articulo en la pagina web de USC Marketing Digital que podria perfectamente responder al ejercicio dos y al tres.

Por una parte define lo que es el BigData:

> "Podemos definirlo como un conjunto de datos cuyo tamaño se encuentra por encima de la capacidad de las herramientas de bases de datos típicas, utilizadas para capturar, almacenar, administrar y analizar la información."

Por otra parte da ejemplos concretos de cuanta información se genera en la diferentes redes sociales que manejamos día a día:

> "Diariamente en Facebook se generan 1.000 millones de comentarios, 300h de videos nuevos por minuto en YouTube, 40.000 búsquedas por segundo en Google y más de 12 Terabytes de tuits diarios."

Ademas de esto, el artículo habla también sobre el tratamiento y almacenaje de estos datos y que finalidad tiene esta tarea.